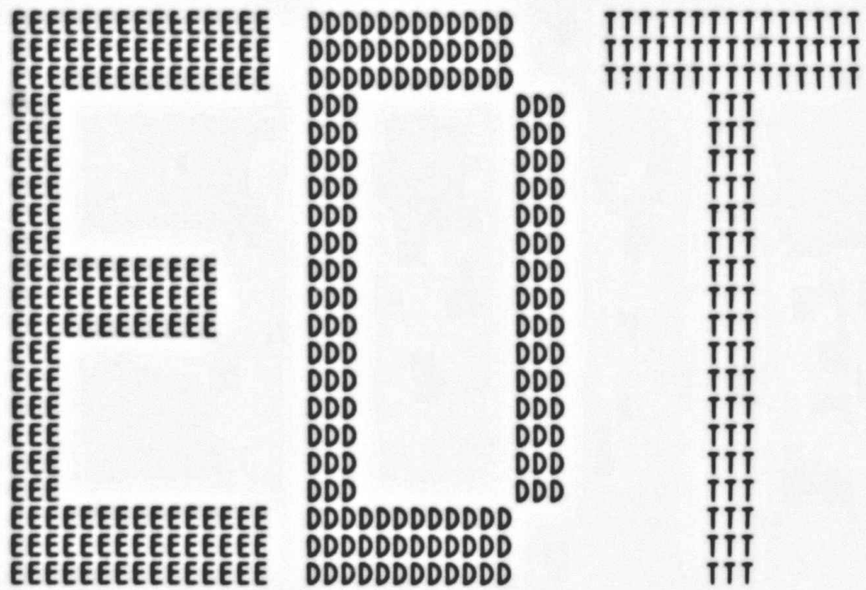
EEEEEEEEEEEEEEE DDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEEEEE DDDDDDDDDDD TTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTT

FILEID**LPRINT

```
LL          PPPPPPPP    RRRRRRRR     IIIIII   NN      NN  TTTTTTTTTT
LL          PPPPPPPP    RRRRRRRR     IIIIII   NN      NN  TTTTTTTTTT
LL          PP      PP  RR      RR     II     NN      NN      TT
LL          PP      PP  RR      RR     II     NN      NN      TT
LL          PP      PP  RR      RR     II     NNNN    NN      TT
LL          PP      PP  RR      RR     II     NNNN    NN      TT
LL          PPPPPPPP    RRRRRRRR       II     NN  NN  NN      TT
LL          PPPPPPPP    RRRRRRRR       II     NN  NN  NN      TT
LL          PP          RR  RR         II     NN    NNNN      TT
LL          PP          RR  RR         II     NN    NNNN      TT
LL          PP          RR    RR       II     NN      NN      TT        ....
LL          PP          RR    RR       II     NN      NN      TT        ....
LLLLLLLLLL  PP          RR      RR   IIIIII   NN      NN      TT        ....
LLLLLLLLLL  PP          RR      RR   IIIIII   NN      NN      TT        ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
;  1          0001  0 %TITLE 'EDT$LPRINT - PRINT line-mode command'
;  2          0002  0 MODULE EDT$LPRINT (                             ! PRINT line-mode command
;  3          0003  0                 IDENT = 'V04-000'               ! File: LPRINT.BLI Edit: JBS1027
;  4          0004  0                 ) =
;  5          0005  1 BEGIN
;  6          0006  1 !
;  7          0007  1 !*****************************************************************************
;  8          0008  1 !*                                                                           *
;  9          0009  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
; 10          0010  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
; 11          0011  1 !*  ALL RIGHTS RESERVED.                                                     *
; 12          0012  1 !*                                                                           *
; 13          0013  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
; 14          0014  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
; 15          0015  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
; 16          0016  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
; 17          0017  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
; 18          0018  1 !*  TRANSFERRED.                                                             *
; 19          0019  1 !*                                                                           *
; 20          0020  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
; 21          0021  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
; 22          0022  1 !*  CORPORATION.                                                             *
; 23          0023  1 !*                                                                           *
; 24          0024  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
; 25          0025  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
; 26          0026  1 !*                                                                           *
; 27          0027  1 !*                                                                           *
; 28          0028  1 !*****************************************************************************
; 29          0029  1 !
; 30          0030  1
; 31          0031  1 !++
; 32          0032  1 ! FACILITY:     EDT -- The DEC Standard Editor
; 33          0033  1 !
; 34          0034  1 ! ABSTRACT:
; 35          0035  1 !
; 36          0036  1 !       This module is called to produce a file containing
; 37          0037  1 !       a specified range of text in a special format.
; 38          0038  1 !
; 39          0039  1 ! ENVIRONMENT:  Runs at any access mode - AST reentrant
; 40          0040  1 !
; 41          0041  1 ! AUTHOR: Bob Kushlis, CREATION DATE: February 3, 1978
; 42          0042  1 !
; 43          0043  1 ! MODIFIED BY:
; 44          0044  1 !
; 45          0045  1 ! 1-001 - Original.  DJS 30-JAN-81.  This module was created by
; 46          0046  1 !       extracting the routines PRINT and EDT$$PRNT_CMD  from EXEC.BLI.
; 47          0047  1 ! 1-002 - Regularize headers.  JBS 20-Mar-1981
; 48          0048  1 ! 1-003 - Use new message codes.  JBS 04-Aug-1981
; 49          0049  1 ! 1-004 - Convert to fileio for reads and writes. STS 15-Jan-1982
; 50          0050  1 ! 1-005 - Pass RHB address to callfio. STS 21-Jan-1982
; 51          0051  1 ! 1-006 - Don't pass descriptors to close file. STS 10-Feb-1982
; 52          0052  1 ! 1-007 - Pass file name to edt$$fiopn_err. STS 25-Feb-1982
; 53          0053  1 ! 1-008 - Add literals for callable EDT. STS 08-Mar-1982
; 54          0054  1 ! 1-009 - Avoid infinitely recursive calls to PRINT.  JBS 11-Mar-1982
; 55          0055  1 ! 1-010 - Print a message on CLOSE errors.  JBS 12-Apr-1982
; 56          0056  1 ! 1-011 - Check for CNTRL/C.  SMB 14-Apr-1982
; 57          0057  1 ! 1-012 - Move conversion to UPCASE for PDP-11's to FILEIO.  SMB 21-Apr-1982
```

```
;    58          0058  1 ! 1-013 - Set a flag if control C actually aborts something.  JBS 24-May-1982
;    59          0059  1 ! 1-014 - Remove reference to SET_FMTWRRUT.  SMB 11-Jun-1982
;    60          0060  1 ! 1-015 - Save buffer position and restore after print. STS 14-Jun-1982
;    61          0061  1 ! 1-016 - Pass default file name in RHB parameter.  JBS 15-Jun-1982
;    62          0062  1 ! 1-017 - Remove EDT$$OPN_FI, EDT$$WR_OFI and EDT$$CLS_FI external references:
;    63          0063  1 !           they are unused.  JBS 15-Jun-1982
;    64          0064  1 ! 1-018 - Stop the working message before second CTRL/C check.  SMB 22-Jun-1982
;    65          0065  1 ! 1-019 - Stop processing on bad select range.  SMB 01-Jul-1982
;    66          0066  1 ! 1-020 - Errors on select must be caught at a higher level.  SMB 02-Jul-1982
;    67          0067  1 ! 1-021 - Change print file message names.  SMB 13-Jul-1982
;    68          0068  1 ! 1-022 - Make edt$$tst_eob in line. STS 22-Sep-1982
;    69          0069  1 ! 1-023 - Make EDT$$RNG_POSFRST in line. STS 11-Oct-1982
;    70          0070  1 ! 1-024 - Reject lines starting with ESC.  JBS 19-Oct-1982
;    71          0071  1 ! 1-025 - Don't use STR$COPY for puts. STS 10-Nov-1982
;    72          0072  1 ! 1-026 - Use LIB$LP_LINES to compute the number of lines per page on VMS.  JBS 29-Apr-1983
;    73          0073  1 ! 1-027 - Correct an error in a comment.  JBS 14-Jun-1983
;    74          0074  1 !--
;    75          0075  1
```

```
;    77          0076  1 %SBTTL 'Declarations'
;    78          0077  1 !
;    79          0078  1 ! TABLE OF CONTENTS:
;    80          0079  1 !
;    81          0080  1
;    82          0081  1 REQUIRE 'EDTSRC:TRAROUNAM';
;    83          0520  1
;    84          0521  1 FORWARD ROUTINE
;    85          0522  1     PRINT,                          ! Format write routine for PRINT command
;    86          0523  1     EDT$$PRNT_CMD : NOVALUE;        ! Process the PRINT command
;    87          0524  1
;    88          0525  1 !
;    89          0526  1 ! INCLUDE FILES:
;    90          0527  1 !
;    91          0528  1
;    92          0529  1 REQUIRE 'EDTSRC:EDTREQ';
;    93          0664  1
;    94        L 0665  1 %IF %BLISS (BLISS32)
;    95          0666  1 %THEN
;    96          0667  1
;    97          0668  1 REQUIRE 'EDTSRC:SYSSYM';
;    98          0698  1
;    99          0699  1 %FI
;   100          0700  1
;   101          0701  1 !
;   102          0702  1 ! MACROS:
;   103          0703  1 !
;   104          0704  1 !       NONE
;   105          0705  1 !
;   106          0706  1 ! EQUATED SYMBOLS:
;   107          0707  1 !
;   108          0708  1
;   109          0709  1 LITERAL
;   110          0710  1     K_MAX_LINES = 66,               ! Max lines per page for PDP-11 systems
;   111          0711  1     K_OVERHEAD_LINES = 6;           ! Number of overhead lines
;   112          0712  1
;   113          0713  1 EXTERNAL LITERAL
;   114          0714  1     EDT$K_PUT,
;   115          0715  1     EDT$K_CLOSE,
;   116          0716  1     EDT$K_CLOSE_DEL,
;   117          0717  1     EDT$K_WRITE_FILE,
;   118          0718  1     EDT$K_OPEN_OUTPUT_NOSEQ;
;   119          0719  1
;   120          0720  1 !
;   121          0721  1 ! OWN STORAGE:
;   122          0722  1 !
;   123          0723  1 !       NONE
;   124          0724  1 !
;   125          0725  1 ! EXTERNAL REFERENCES:
;   126          0726  1 !
;   127          0727  1 !       In the routine
```

```
;   129          0728  1 %SBTTL 'PRINT - intercept formatted output'
;   130          0729  1 ROUTINE PRINT (                                      ! Intercept formatted output
;   131          0730  1     RECADDR,                                          ! Address of record
;   132          0731  1     RECLEN                                            ! Length of record
;   133          0732  1     ) =
;   134          0733  1
;   135          0734  1 !++
;   136          0735  1 ! FUNCTIONAL DESCRIPTION:
;   137          0736  1 !
;   138          0737  1 !       This routine is made the format write routine when doing a PRINT
;   139          0738  1 !       command.  Whenever the formatting routines are to output a record
;   140          0739  1 !       this routine is called, which in turn writes the line to the file.
;   141          0740  1 !
;   142          0741  1 ! FORMAL PARAMETERS:
;   143          0742  1 !
;   144          0743  1 !  RECADDR              Address of the record to write
;   145          0744  1 !
;   146          0745  1 !  RECLEN               Length of that record
;   147          0746  1 !
;   148          0747  1 ! IMPLICIT INPUTS:
;   149          0748  1 !
;   150          0749  1 !       NONE
;   151          0750  1 !
;   152          0751  1 ! IMPLICIT OUTPUTS:
;   153          0752  1 !
;   154          0753  1 !       NONE
;   155          0754  1 !
;   156          0755  1 ! ROUTINE VALUE:
;   157          0756  1 !
;   158          0757  1 !       Same as EDT$$WR_OFI
;   159          0758  1 !
;   160          0759  1 ! SIDE EFFECTS:
;   161          0760  1 !
;   162          0761  1 !       Changes the formatted write routine to EDT$$TI_WRLN during
;   163          0762  1 !       I/O, then restores it before returning.
;   164          0763  1 !
;   165          0764  1 !--
;   166          0765  1
;   167          0766  2     BEGIN
;   168          0767  2
;   169          0768  2     EXTERNAL ROUTINE
;   170          0769  2         EDT$$CALLFIO,
;   171          0770  2         EDT$$TI_WRLN;
;   172          0771  2
;   173          0772  2     EXTERNAL
;   174          0773  2         EDT$$A_CUR_BUF : REF TBCB_BLOCK,
;   175          0774  2         EDT$$A_FMT_WRRUT,
;   176          0775  2         EDT$$L_IO_VFCHD;
;   177          0776  2
;   178          0777  2     LOCAL
;   179          0778  2         STATUS,
;   180          0779  2         LEN,
;   181          0780  2         ADDR,
;   182          0781  2         RHB_DESC : BLOCK [8, BYTE],
;   183          0782  2         FILE_DESC : BLOCK [8, BYTE];
;   184          0783  2
;   185        L 0784  2 %IF %BLISS (BLISS32)
```

```
;  186          0785  2 %THEN
;  187          0786  2     RHB_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
;  188          0787  2     RHB_DESC [DSC$B_CLASS] = DSC$K_CLASS_S;
;  189          0788  2     FILE_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
;  190          0789  2     FILE_DESC [DSC$B_CLASS] = DSC$K_CLASS_S;
;  191          0790  2 %FI
;  192          0791  2
;  193          0792  2     FILE_DESC [DSC$A_POINTER] = .RECADDR;
;  194          0793  2     FILE_DESC [DSC$W_LENGTH] = .RECLEN;
;  195          0794  2     RHB_DESC [DSC$A_POINTER] = EDT$$L_IO_VFCHD;
;  196          0795  2 !+
;  197          0796  2 ! Just in case the I/O routines have an error and decide to
;  198          0797  2 ! print a message about it, set the format write routine
;  199          0798  2 ! back to EDT$$TI_WRLN for the duration of the I/O.
;  200          0799  2 !-
;  201          0800  2     EDT$$A_FMT_WRRUT = EDT$$TI_WRLN;
;  202          0801  2 !+
;  203          0802  2 ! Reject any lines that start with ESC.  This is because if we do a PRINT in
;  204          0803  2 ! change mode, EDT will try to put the text at the bottom of the screen, and
;  205          0804  2 ! will issue escape sequences to this effect.  Since we don't want these
;  206          0805  2 ! escape sequences to go into the file, reject them.
;  207          0806  2 !-
;  208          0807  2
;  209          0808  3     IF ((CH$RCHAR (.RECADDR) EQL ASC_K_ESC) AND .RECLEN NEQ 0)  !
;  210          0809  2     THEN
;  211          0810  2         STATUS = 1
;  212          0811  2     ELSE
;  213          0812  2         STATUS = EDT$$CALLFIO (EDT$K_PUT, EDT$K_WRITE_FILE, FILE_DESC, RHB_DESC);
;  214          0813  2
;  215          0814  2 !+
;  216          0815  2 ! Now restore this routine as the formatted write routine.
;  217          0816  2 !-
;  218          0817  2     EDT$$A_FMT_WRRUT = PRINT;
;  219          0818  2     RETURN (.STATUS);
;  220          0819  1     END;                                    ! of routine PRINT


                                                .TITLE  EDT$LPRINT EDT$LPRINT - PRINT line-mode command
                                                .IDENT  \V04-000\

                                                .EXTRN  EDT$K_PUT, EDT$K_CLOSE
                                                .EXTRN  EDT$K_CLOSE_DEL
                                                .EXTRN  EDT$K_WRITE_FILE
                                                .EXTRN  EDT$K_OPEN_OUTPUT_NOSEQ
                                                .EXTRN  EDT$$CALLFIO, EDT$$TI_WRLN
                                                .EXTRN  EDT$$A_CUR_BUF, EDT$$A_FMT_WRRUT
                                                .EXTRN  EDT$$L_IO_VFCHD

                                                .PSECT  _EDT$CODE,NOWRT,  SHR,  PIC,2

                                0004 00000 PRINT:  .WORD   Save R2                                  ; 0729
                52 00000000G     00  9E 00002       MOVAB   EDT$$A_FMT_WRRUT, R2                   ;
                5E               10  C2 00009       SUBL2   #16, SP                                ;
          0A    AE      010E     8F  B0 0000C       MOVW    #270, RHB_DESC+2                       ; 0786
          02    AE      010E     8F  B0 00012       MOVW    #270, FILE_DESC+2                      ; 0788
          04    AE        04     AC  D0 00018       MOVL    RECADDR, FILE_DESC+4                   ; 0792
                6E        08     AC  B0 0001D       MOVW    RECLEN, FILE_DESC                      ; 0793
```

```
                         0C   AE 00000000G   00   9E 00021          MOVAB   EDT$$L_IO_VFCHD, RHB_DESC+4                ; 0794
                              62 00000000G   00   9E 00029          MOVAB   EDT$$TI_WRLN, EDT$$A_FMT_WRRUT           ; 0800
                              1B       04    BC   91 00030          CMPB    @RECADDR, #27                            ; 0808
                                             0A   12 00034          BNEQ    1$                                       ;
                                       08    AC   D5 00036          TSTL    RECLEN                                   ;
                                             05   13 00039          BEQL    1$                                       ;
                                 50          01   D0 0003B          MOVL    #1, STATUS                               ; 0810
                                             19   11 0003E          BRB     2$                                       ;
                                       08    AE   9F 00040 1$:      PUSHAB  RHB_DESC                                 ; 0812
                                       04    AE   9F 00043          PUSHAB  FILE_DESC                                ;
                                00000000G    8F   DD 00046          PUSHL   #EDT$K_WRITE_FILE                        ;
                                00000000G    8F   DD 0004C          PUSHL   #EDT$K_PUT                               ;
                    00000000G   00            04   FB 00052          CALLS   #4, EDT$$CALLFIO                         ;
                              62       A4    AF   9E 00059 2$:      MOVAB   PRINT, EDT$$A_FMT_WRRUT                  ; 0817
                                             04   0005D              RET                                              ; 0819

; Routine Size:  94 bytes,    Routine Base:  _EDT$CODE + 0000
```

```
;  222     0820  1 %SBTTL 'EDT$$PRNT_CMD  - PRINT line-mode command'
;  223     0821  1
;  224     0822  1 GLOBAL ROUTINE EDT$$PRNT_CMD                    ! PRINT line-mode command
;  225     0823  1     : NOVALUE =
;  226     0824  1
;  227     0825  1 !++
;  228     0826  1 ! FUNCTIONAL DESCRIPTION:
;  229     0827  1 !
;  230     0828  1 !       Command processing for PRINT.  First, attempt to open the file.
;  231     0829  1 !       If it succeeds then set up the little routine above as the format
;  232     0830  1 !       write routine and process the range.  A page skip is done after
;  233     0831  1 !       60 lines, or when the first character of a line is a form feed,
;  234     0832  1 !       whichever comes first.
;  235     0833  1 !
;  236     0834  1 ! FORMAL PARAMETERS:
;  237     0835  1 !
;  238     0836  1 !       NONE
;  239     0837  1 !
;  240     0838  1 ! IMPLICIT INPUTS:
;  241     0839  1 !
;  242     0840  1 !       EDT$$A_CUR_BUF
;  243     0841  1 !       EDT$$G_TXT_ONSCR
;  244     0842  1 !       EDT$$G_TI_WID
;  245     0843  1 !       EDT$$Z_RNG_ORIGPOS
;  246     0844  1 !       EDT$$A_WK_LN
;  247     0845  1 !       EDT$$A_EXE_CURCMD
;  248     0846  1 !
;  249     0847  1 ! IMPLICIT OUTPUTS:
;  250     0848  1 !
;  251     0849  1 !       EDT$$G_TXT_ONSCR
;  252     0850  1 !       EDT$$G_CC_DONE
;  253     0851  1 !
;  254     0852  1 ! ROUTINE VALUE:
;  255     0853  1 !
;  256     0854  1 !       NONE
;  257     0855  1 !
;  258     0856  1 ! SIDE EFFECTS:
;  259     0857  1 !
;  260     0858  1 !       On exit from this routine, the formatting routine is set to EDT$$TI_WRLN.
;  261     0859  1 !       While it is running, the formatting routine is usually in this module.
;  262     0860  1 !
;  263     0861  1 !--
;  264     0862  1
;  265     0863  2     BEGIN
;  266     0864  2
;  267     0865  2     EXTERNAL ROUTINE
;  268     0866  2         EDT$$STOP_WKINGMSG,
;  269     0867  2         EDT$$CHK_CC,
;  270     0868  2         EDT$$CALL_FIO,
;  271     0869  2         EDT$$OUT_FMTBUF,
;  272     0870  2         EDT$$FMT_MSG,
;  273     0871  2         EDT$$TI_WRLN,
;  274     0872  2         EDT$$NXT_LNRNG,
;  275     0873  2         EDT$$RNG_REPOS,
;  276     0874  2         EDT$$RD_CURLN,
;  277     0875  2         EDT$$FIOPN_ERR,
;  278     0876  2         EDT$$TY_CURLN;
```

```
;  279          0877  2
;  280        L 0878  2 %IF %BLISS (BLISS32)
;  281          0879  2 %THEN
;  282          0880  2
;  283          0881  2     EXTERNAL ROUTINE
;  284          0882  2         STR$FREE1_DX,                       ! needed for deallocation of dynamic descriptors
;  285          0883  2         LIB$LP_LINES;                       ! Compute number of lines per page
;  286          0884  2
;  287          0885  2 %FI
;  288          0886  2
;  289          0887  2     EXTERNAL
;  290          0888  2         EDT$$G_EXT_MOD,                     ! Are we in EXT mode
;  291          0889  2         EDT$$A_FMT_WRRUT,                   ! Address of write routine
;  292          0890  2         EDT$$Z_RNG_SAVPOS,
;  293          0891  2         EDT$$A_FMT_CUR,
;  294          0892  2         EDT$$T_FMT_BUF,
;  295          0893  2         EDT$$A_CUR_BUF : REF TBCB_BLOCK,
;  296          0894  2         EDT$$G_TXT_ONSCR,
;  297          0895  2         EDT$$G_TI_WID,
;  298          0896  2         EDT$$Z_RNG_ORIGPOS : POS_BLOCK,
;  299          0897  2         EDT$$Z_EOB_LN,
;  300          0898  2         EDT$$G_RNG_FRSTLN,
;  301          0899  2         EDT$$A_WK_LN : REF LIN_BLOCK,
;  302          0900  2         EDT$$A_EXE_CURCMD : REF NODE_BLOCK, ! Pointer to the current command.
;  303          0901  2         EDT$$G_CC_DONE;                     ! Set to 1 if control C actually aborted something
;  304          0902  2
;  305          0903  2     MESSAGES ((NOFILSPC, PRIFILCRE, PRIFILCLO));
;  306          0904  2
;  307          0905  2     LOCAL
;  308          0906  2         FORMAT_ROUTINE,                     ! Save the format routine entered with
;  309          0907  2         COUNT,                              ! Number of lines on this page
;  310          0908  2         FILE_DESC : BLOCK [8, BYTE],
;  311          0909  2         RHB_DESC : BLOCK [8, BYTE],
;  312          0910  2         SAV_BUF,
;  313          0911  2         STRLEN,
;  314          0912  2         STRADR,
;  315          0913  2         SAVE_WIDTH,                         ! Save the terminal width
;  316          0914  2         IFI,                                ! IFI of the output file.
;  317          0915  2         MAX_LINES;                          ! Number of lines par page
;  318          0916  2
;  319          0917  2     BIND
;  320          0918  2         RAN = .EDT$$A_EXE_CURCMD [RANGE1] : NODE_BLOCK;
;  321          0919  2
;  322          0920  2 !+
;  323          0921  2 ! Make sure there is a file spec.
;  324          0922  2 !-
;  325          0923  2
;  326          0924  3     IF (.EDT$$A_EXE_CURCMD [FSPCLEN] EQL 0)
;  327          0925  2     THEN
;  328          0926  3         BEGIN
;  329          0927  3         EDT$$FMT_MSG (EDT$_NOFILSPC);
;  330          0928  3         RETURN;
;  331          0929  2         END;
;  332          0930  2
;  333          0931  2 !+
;  334          0932  2 ! If the range is null, then make it the whole buffer.
;  335          0933  2 !-
```

```
;  336          0934  2      SAV_BUF = .EDT$$A_CUR_BUF;                      ! save original address
;  337          0935  2
;  338          0936  2      IF (.RAN [RAN_TYPE] EQL RAN_NULL) THEN RAN [RAN_TYPE] = RAN_WHOLE;
;  339          0937  2
;  340          0938  2  !+
;  341          0939  2  ! Position to top of range.
;  342          0940  2  !-
;  343          0941  2      EDT$$G_RNG_FRSTLN = 1;
;  344          0942  2      EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_ORIGPOS);
;  345          0943  2
;  346          0944  2      IF ( NOT EDT$$RNG_REPOS (.EDT$$A_EXE_CURCMD [RANGE1])) THEN RETURN;
;  347          0945  2
;  348          0946  2      FORMAT_ROUTINE = .EDT$$A_FMT_WRRUT;
;  349          0947  2  !+
;  350          0948  2  ! Compute the number of lines per page.
;  351          0949  2  !-
;  352          0950  2
;  353        L 0951  2  %IF %BLISS (BLISS32)
;  354          0952  2  %THEN
;  355          0953  2      MAX_LINES = MAX (1, LIB$LP_LINES () - K_OVERHEAD_LINES);
;  356        U 0954  2  %ELSE
;  357        U 0955  2      MAX_LINES = MAX (1, K_MAX_LINES - K_OVERHEAD_LINES);
;  358          0956  2  %FI
;  359          0957  2
;  360          0958  2  !+
;  361          0959  2  ! Set up so a form feed will be output immediately.
;  362          0960  2  !-
;  363          0961  2      COUNT = .MAX_LINES;
;  364          0962  2  !+
;  365          0963  2  ! Open the file.
;  366          0964  2  !-
;  367          0965  2
;  368        L 0966  2  %IF %BLISS (BLISS32)
;  369          0967  2  %THEN
;  370          0968  2      FILE_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
;  371          0969  2      FILE_DESC [DSC$B_CLASS] = DSC$K_CLASS_D;
;  372          0970  2      RHB_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
;  373          0971  2      RHB_DESC [DSC$B_CLASS] = DSC$K_CLASS_D;
;  374          0972  2  %FI
;  375          0973  2
;  376          0974  2      FILE_DESC [DSC$A_POINTER] = 0;
;  377          0975  2      FILE_DESC [DSC$W_LENGTH] = 0;
;  378          0976  2      RHB_DESC [DSC$A_POINTER] = 0;
;  379          0977  2      RHB_DESC [DSC$W_LENGTH] = 0;
;  380          0978  2      STRLEN = .EDT$$A_EXE_CURCMD [FSPCLEN];
;  381          0979  2      STRADR = .EDT$$A_EXE_CURCMD [FILSPEC];
;  382          0980  2      STRING_DESC (FILE_DESC, STRLEN, .STRADR);
;  383          0981  2      IFI = EDT$$CALLFIO (EDT$K_OPEN_OUTPUT_NOSEQ, EDT$K_WRITE_FILE, FILE_DESC, RHB_DESC);
;  384          0982  2
;  385          0983  3      IF (.IFI NEQ 0)
;  386          0984  2      THEN
;  387          0985  3          BEGIN
;  388          0986  3  !+
;  389          0987  3  ! Save the current terminal width and make it 132 characters for the printer.
;  390          0988  3  !-
;  391          0989  3          SAVE_WIDTH = .EDT$$G_TI_WID;
;  392          0990  3          EDT$$G_TI_WID = 132;
```

```
;  393          0991  3 !+
;  394          0992  3 ! Reset the format writing routine.
;  395          0993  3 !-
;  396          0994  3         EDT$$A_FMT_WRRUT = PRINT;
;  397          0995  3 !+
;  398          0996  3 ! Loop through the range.
;  399          0997  3 !-
;  400          0998  3
;  401          0999  3         WHILE (EDT$$NXT_LNRNG (0) AND ( NOT EDT$$CHK_CC ())) DO
;  402          1000  4             BEGIN
;  403          1001  4
;  404          1002  5             IF (.EDT$$A_WK_LN NEQA EDT$$Z_EOB_LN)
;  405          1003  4             THEN
;  406          1004  5                 BEGIN
;  407          1005  5 !+
;  408          1006  5 ! Look for form-feed in the record.
;  409          1007  5 !-
;  410          1008  5
;  411          1009  5                 IF (CH$RCHAR (EDT$$A_WK_LN [LIN_TEXT]) EQL 12) THEN COUNT = .MAX_LINES;
;  412          1010  5
;  413          1011  5 !+
;  414          1012  5 ! Check for a page skip.
;  415          1013  5 !-
;  416          1014  5
;  417          1015  6                 IF (.COUNT EQL .MAX_LINES)
;  418          1016  5                 THEN
;  419          1017  6                     BEGIN
;  420          1018  6                     PRINT (UPLIT BYTE(12), 1);
;  421          1019  6 !+
;  422          1020  6 ! now dump out two blank lines
;  423          1021  6 ! first make sure that the format buffer is empty
;  424          1022  6 !-
;  425          1023  6                     EDT$$A_FMT_CUR = EDT$$T_FMT_BUF;
;  426          1024  6                     EDT$$OUT_FMTBUF ();
;  427          1025  6                     EDT$$OUT_FMTBUF ();
;  428          1026  6                     COUNT = 0;
;  429          1027  5                     END;
;  430          1028  5
;  431          1029  5                 COUNT = .COUNT + 1;
;  432          1030  5 !+
;  433          1031  5 ! And print the line.
;  434          1032  5 !-
;  435          1033  5
;  436          1034  5                 IF ( NOT EDT$$TY_CURLN ()) THEN EXITLOOP;
;  437          1035  5
;  438          1036  4                 END;
;  439          1037  4
;  440          1038  3             END;
;  441          1039  3
;  442          1040  3         IF (.EDT$$G_EXT_MOD) THEN EDT$$STOP_WKINGMSG ();
;  443          1041  3
;  444          1042  3 !+
;  445          1043  3 ! Reposition to the first line of the range.
;  446          1044  3 !-
;  447          1045  3         EDT$$CPY_MEM (POS_SIZE, EDT$$Z_RNG_SAVPOS, .EDT$$A_CUR_BUF);
;  448          1046  3         EDT$$RD_CURLN ();
;  449          1047  3 !+
```

```
;  450          1048  3 ! Close the file.
;  451          1049  3 !-
;  452          1050  3
;  453          1051  3         IF EDT$$CHK_CC ()
;  454          1052  3         THEN
;  455          1053  4             BEGIN
;  456          1054  4
;  457          1055  5             IF ( NOT EDT$$CALLFIO (EDT$K_CLOSE_DEL, EDT$K_WRITE_FILE, FILE_DESC, 0))
;  458          1056  4             THEN
;  459          1057  4                 EDT$$FIOPN_ERR (EDT$_PRIFILCRE, FILE_DESC);
;  460          1058  4
;  461          1059  4             EDT$$G_CC_DONE = 1;
;  462          1060  4             END
;  463          1061  3         ELSE
;  464          1062  3
;  465          1063  4             IF ( NOT EDT$$CALLFIO (EDT$K_CLOSE, EDT$K_WRITE_FILE, FILE_DESC, 0))
;  466          1064  3             THEN
;  467          1065  3                 EDT$$FIOPN_ERR (EDT$_PRIFILCRE, FILE_DESC);
;  468          1066  3
;  469          1067  3 !+
;  470          1068  3 ! Restore the terminal width and the format write routine,
;  471          1069  3 ! and reposition to the original line.
;  472          1070  3 !-
;  473          1071  3         EDT$$G_TI_WID = .SAVE_WIDTH;
;  474          1072  3         EDT$$A_FMT_WRRUT = EDT$$TI_WRLN;
;  475          1073  3         EDT$$A_CUR_BUF = .SAV_BUF;              ! first get the buffer address
;  476          1074  3         EDT$$CPY_MEM (POS_SIZE, EDT$$Z_RNG_ORIGPOS, .EDT$$A_CUR_BUF);
;  477          1075  3         EDT$$RD_CURLN ();
;  478          1076  3         END
;  479          1077  2     ELSE
;  480          1078  2 !+
;  481          1079  2 ! Here if file was not opened.
;  482          1080  2 !-
;  483          1081  2         EDT$$FIOPN_ERR (EDT$_PRIFILCRE, FILE_DESC);
;  484          1082  2
;  485          1083  2     EDT$$A_FMT_WRRUT = .FORMAT_ROUTINE;
;  486          1084  2 !+
;  487          1085  2 ! On VMS deallocate dynamic descriptors
;  488          1086  2 !-
;  489          1087  2
;  490        L 1088  2 %IF %BLISS (BLISS32)
;  491          1089  2 %THEN
;  492          1090  2     STR$FREE1_DX (FILE_DESC);
;  493          1091  2     STR$FREE1_DX (RHB_DESC);
;  494          1092  2 %FI
;  495          1093  2
;  496          1094  1     END;                                    ! of routine EDT$$PRNT_CMD


                                        0C  0005E P.AAA:  .BYTE   12                                                    ;

                                                          .EXTRN  EDT$$STOP_WKINGMSG
                                                          .EXTRN  EDT$$CHK_CC, EDT$$OUT_FMTBUF
                                                          .EXTRN  EDT$$FMT_MSG, EDT$$NXT_LNRNG
                                                          .EXTRN  EDT$$RNG_REPOS, EDT$$RD_CURLN
                                                          .EXTRN  EDT$$FIOPN_ERR, EDT$$TY_CURLN
```

```
                                                        .EXTRN  STR$FREE1_DX, LIB$LP_LINES
                                                        .EXTRN  EDT$$G_EXT_MOD, EDT$$Z_RNG_SAVPOS
                                                        .EXTRN  EDT$$A_FMT_CUR, EDT$$T_FMT_BUF
                                                        .EXTRN  EDT$$G_TXT_ONSCR
                                                        .EXTRN  EDT$$G_TI_WID, EDT$$Z_RNG_ORIGPOS
                                                        .EXTRN  EDT$$Z_EOB_LN, EDT$$G_RNG_FRSTLN
                                                        .EXTRN  EDT$$A_WK_LN, EDT$$A_EXE_CURCMD
                                                        .EXTRN  EDT$$G_CC_DONE, EDT$_NOFILSPC
                                                        .EXTRN  EDT$_PRIFILCRE, EDT$_PRIFILCLO
                                                        .EXTRN  STR$COPY_R

                                        0FFC 00000      .ENTRY  EDT$$PRNT_CMD, Save R2,R3,R4,R5,R6,R7,R8,-  ; 0822
                                                                R9,R10,R11
                                5B 00000000G   8F   D0 00002    MOVL    #EDT$K_WRITE_FILE, R11
                                5A 00000000G   00   9E 00009    MOVAB   EDT$$A_FMT_WRRUT, R10
                                59 00000000G   00   9E 00010    MOVAB   EDT$$A_CUR_BUF, R9
                                5E                  14   C2 00017    SUBL2   #20, SP
                                50 00000000G   00   D0 0001A    MOVL    EDT$$A_EXE_CURCMD, R0               ; 0918
                                52                04   A0 D0 00021    MOVL    4(R0), R2
                                                  0C   A0 D5 00025    TSTL    12(R0)                          ; 0924
                                                       0E   12 00028    BNEQ    1$
                                   00000000G   8F   DD 0002A    PUSHL   #EDT$_NOFILSPC                      ; 0927
                     00000000G  00             01   FB 00030    CALLS   #1, EDT$$FMT_MSG
                                                  04   00037    RET                                         ; 0926
                                50                69   D0 00038 1$:     MOVL    EDT$$A_CUR_BUF, R0          ; 0934
                                57                50   D0 0003B    MOVL    R0, SAV_BUF
                                                  01   A2 95 0003E    TSTB    1(R2)                           ; 0936
                                                       04   12 00041    BNEQ    2$
                                      01   A2              0B   90 00043    MOVB    #11, 1(R2)
                                00000000G  00             01   D0 00047 2$:     MOVL    #1, EDT$$G_RNG_FRSTLN       ; 0941
                  00000000G  00                 60   0E   28 0004E    MOVC3   #14, (R0), EDT$$Z_RNG_ORIGPOS   ; 0942
                                50 00000000G   00   D0 00056    MOVL    EDT$$A_EXE_CURCMD, R0               ; 0944
                                                  04   A0 DD 0005D    PUSHL   4(R0)
                                00000000G  00             01   FB 00060    CALLS   #1, EDT$$RNG_REPOS
                                00000000G  01             50   E8 00067    BLBS    R0, 3$
                                                  04   00067    RET                                         ; 0946
                                58                6A   D0 0006B 3$:     MOVL    EDT$$A_FMT_WRRUT, FORMAT_ROUTINE    ; 0953
                                00000000G  00             00   FB 0006E    CALLS   #0, LIB$LP_LINES
                                50                06   C2 00075    SUBL2   #6, R0
                                                  03   14 00078    BGTR    4$
                                50                01   D0 0007A    MOVL    #1, R0
                                53                50   D0 0007D 4$:     MOVL    R0, MAX_LINES
                                52                53   D0 00080    MOVL    MAX_LINES, COUNT                ; 0961
                          0C    AE 020E0000       8F   D0 00083    MOVL    #34471936, FILE_DESC            ; 0975
                          04    AE 020E0000       8F   D0 0008B    MOVL    #34471936, RHB_DESC             ; 0977
                                                  10   AE D4 00093    CLRL    FILE_DESC+4                     ; 0974
                                                  08   AE D4 00096    CLRL    RHB_DESC+4                      ; 0976
                                50 00000000G   00   D0 00099    MOVL    EDT$$A_EXE_CURCMD, R0               ; 0978
                                6E                0C   A0 D0 000A0    MOVL    12(R0), STRLEN
                                50                08   A0 D0 000A4    MOVL    8(R0), STRADR                   ; 0979
                                                       50   DD 000A8    PUSHL   STRADR                      ; 0980
                                                  04   AE 9F 000AA    PUSHAB  STRLEN
                                                  14   AE 9F 000AD    PUSHAB  FILE_DESC
                                00000000G  00             03   FB 000B0    CALLS   #3, STR$COPY_R
                                                  04   AE 9F 000B7    PUSHAB  RHB_DESC                    ; 0981
                                                  10   AE 9F 000BA    PUSHAB  FILE_DESC
                                                       5B   DD 000BD    PUSHL   R11
```

```
                                00000000G  8F   DD 000BF          PUSHL   #EDT$K_OPEN_OUTPUT_NOSEQ                          ;
                      00000000G  00        04   FB 000C5          CALLS   #4, EDT$$CALLFIO                                   ;
                                           50   D5 000CC          TSTL    IFI                                                ; 0983
                                           03   12 000CE          BNEQ    5$                                                 ;
                                        0123   31 000D0          BRW     14$                                                ;
                                56 00000000G  00   D0 000D3 5$:    MOVL    EDT$$G_TI_WID, SAVE_WIDTH                          ; 0989
                      00000000G  00        84   8F   9A 000DA    MOVZBL  #132, EDT$$G_TI_WID                                ; 0990
                                6A      FEBB   CF   9E 000E2    MOVAB   PRINT, EDT$$A_FMT_WRRUT                            ; 0994
                                           7E   D4 000E7 6$:    CLRL    -(SP)                                              ; 0999
                      00000000G  00           01   FB 000E9    CALLS   #1, EDT$$NXT_LNRNG                                 ;
                                5D           50   E9 000F0    BLBC    R0, 9$                                             ;
                      00000000G  00           00   FB 000F3    CALLS   #0, EDT$$CHK_CC                                    ;
                                53           50   E8 000FA    BLBS    R0, 9$                                             ;
                                50 00000000G  00   D0 000FD    MOVL    EDT$$A_WK_LN, R0                                   ; 1002
                                51 00000000G  00   9E 00104    MOVAB   EDT$$Z_EOB_LN, R1                                  ;
                                51           50   D1 0010B    CMPL    R0, R1                                             ;
                                           D7   13   0010E    BEQL    6$                                                 ;
                                0C        07   A0   91 00110    CMPB    7(R0), #12                                         ; 1009
                                           03   12   00114    BNEQ    7$                                                 ;
                                52           53   D0 00116    MOVL    MAX_LINES, COUNT                                   ;
                                53           52   D1 00119 7$:    CMPL    COUNT, MAX_LINES                                   ; 1015
                                           26   12   0011C    BNEQ    8$                                                 ;
                                           01   DD   0011E    PUSHL   #1                                                 ; 1018
                                      FEDB   CF   9F 00120    PUSHAB  P.AAA                                              ;
                                   FE78   CF   02   FB 00124    CALLS   #2, PRINT                                          ;
                      00000000G  00 00000000G  00   9E 00129    MOVAB   EDT$$T_FMT_BUF, EDT$$A_FMT_CUR                     ; 1023
                      00000000G  00           00   FB 00134    CALLS   #0, EDT$$OUT_FMTBUF                                ; 1024
                      00000000G  00           00   FB 0013B    CALLS   #0, EDT$$OUT_FMTBUF                                ; 1025
                                           52   D4 00142    CLRL    COUNT                                              ; 1026
                                           52   D6 00144 8$:    INCL    COUNT                                              ; 1029
                      00000000G  00           00   FB 00146    CALLS   #0, EDT$$TY_CURLN                                  ; 1034
                                97           50   E8 0014D    BLBS    R0, 6$                                             ;
                                07 00000000G  00   E9 00150 9$:    BLBC    EDT$$G_EXT_MOD, 10$                                ; 1040
                      00000000G  00           00   FB 00157    CALLS   #0, EDT$$STOP_WKINGMSG                             ;
                                50           69   D0 0015E 10$:   MOVL    EDT$$A_CUR_BUF, R0                                 ; 1045
                           60 00000000G  00   0E   28 00161    MOVC3   #14, EDT$$Z_RNG_SAVPOS, (R0)                       ;
                      00000000G  00           00   FB 00169    CALLS   #0, EDT$$RD_CURLN                                  ; 1046
                      00000000G  00           00   FB 00170    CALLS   #0, EDT$$CHR_CC                                    ; 1051
                                30           50   E9 00177    BLBC    R0, 12$                                            ;
                                           7E   D4 0017A    CLRL    -(SP)                                              ; 1055
                                      10   AE   9F 0017C    PUSHAB  FILE_DESC                                          ;
                                           5B   DD 0017F    PUSHL   R11                                                ;
                                00000000G  8F   DD 00181    PUSHL   #EDT$K_CLOSE_DEL                                   ;
                      00000000G  00           04   FB 00187    CALLS   #4, EDT$$CALLFIO                                   ;
                                10           50   E8 0018E    BLBS    R0, 11$                                            ;
                                      0C   AE   9F 00191    PUSHAB  FILE_DESC                                          ; 1057
                                00000000G  8F   DD 00194    PUSHL   #EDT$_PRIFILCRE                                    ;
                      00000000G  00           02   FB 0019A    CALLS   #2, EDT$$FIOPN_ERR                                 ;
                      00000000G  00           01   D0 001A1 11$:    MOVL    #1, EDT$$G_CC_DONE                                 ; 1059
                                           27   11   001A8    BRB     13$                                                ; 1051
                                           7E   D4   001AA 12$:    CLRL    -(SP)                                              ; 1063
                                      10   AE   9F   001AC    PUSHAB  FILE_DESC                                          ;
                                           5B   DD   001AF    PUSHL   R11                                                ;
                                00000000G  8F   DD   001B1    PUSHL   #EDT$K_CLOSE                                       ;
                      00000000G  00           04   FB   001B7    CALLS   #4, EDT$$CALLFIO                                   ;
                                10           50   E8   001BE    BLBS    R0, 13$                                            ;
                                      0C   AE   9F   001C1    PUSHAB  FILE_DESC                                          ; 1065
```

```
                                00000000G   8F   DD 001C4          PUSHL   #EDT$_PRIFILCRE                                 ;
                     00000000G  00           02   FB 001CA          CALLS   #2, EDT$$FIOPN_ERR                              ;
                     00000000G  00           56   D0 001D1 13$:     MOVL    SAVE_WIDTH, EDT$$G_TI_WID                       ; 1071
                                6A 00000000G 00   9E 001D8          MOVAB   EDT$$TI_WRLN, EDT$$A_FMT_WRRUT                  ; 1072
                                69           57   D0 001DF          MOVL    SAV_BUF, EDT$$A_CUR_BUF                         ; 1073
                                50           69   D0 001E2          MOVL    EDT$$A_CUR_BUF, R0                              ; 1074
                           60 00000000G      00   0E 28 001E5       MOVC3   #14, EDT$$Z_RNG_ORIGPOS, (R0)                   ;
                              00000000G      00   00 FB 001ED       CALLS   #0, EDT$$RD_CURLN                               ; 1075
                                                  10 11 001F4       BRB     15$                                             ; 0983
                                          0C      AE 9F 001F6 14$:  PUSHAB  FILE_DESC                                       ; 1081
                                00000000G         8F DD 001F9       PUSHL   #EDT$_PRIFILCRE                                 ;
                     00000000G  00           02   FB 001FF          CALLS   #2, EDT$$FIOPN_ERR                              ;
                                6A           58   D0 00206 15$:     MOVL    FORMAT_ROUTINE, EDT$$A_FMT_WRRUT                ; 1083
                                          0C      AE 9F 00209       PUSHAB  FILE_DESC                                       ; 1090
                     00000000G  00           01   FB 0020C          CALLS   #1, STR$FREE1_DX                                ;
                                          04      AE 9F 00213       PUSHAB  RHB_DESC                                        ; 1091
                     00000000G  00           01   FB 00216          CALLS   #1, STR$FREE1_DX                                ;
                                                  04 0021D          RET                                                     ; 1094

; Routine Size:  542 bytes,    Routine Base:  _EDT$CODE + 005F

;  497          1095  1
;  498          1096  1 !<BLF/PAGE>
```

```
;  500          1097  1 END                                   ! of module EDT$LPRINT
;  501          1098  1
;  502          1099  0 ELUDOM




;                                PSECT SUMMARY
;
;       Name                    Bytes                   Attributes
;
;  _EDT$CODE                        637  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)


;                              Library Statistics
;                                        --------  Symbols  --------      Pages       Processing
;       File                             Total     Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[EDT.SRC]EDT.L32;1          377         95        25          40        00:00.2
;  _$255$DUA28:[EDT.SRC]PSECTS.L32;1         2          1        50           7        00:00.1
;  _$255$DUA28:[SYSLIB]STARLET.L32;1      9776          7         0         581        00:04.1




;                                   COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS$:LPRINT/OBJ=OBJ$:LPRINT MSRC$:LPRINT.BLI/UPDATE=(ENH$:LPRINT)

; Size:          636 code + 1 data bytes
; Run Time:          00:33.2
; Elapsed Time:      00:43.0
; Lines/CPU Min:      1987
; Lexemes/CPU-Min:   7143
; Memory Used:   181 pages
; Compilation Complete
```

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

LINCL LIS
LQUERY LIS
LSUB LIS
LTYPE LIS
LRES LIS
LTADJ LIS
LPRINT LIS
LGETSTR LIS
LPUTCHR LIS
LSUBSN LIS
LSHOW LIS
LMOVE LIS
LSET LIS
LWRITE LIS
LINSERT LIS
LNONCTG LIS
LSUBS LIS